```
NNN        NNN    CCCCCCCCCCCC    PPPPPPPPPPPP
NNN        NNN    CCCCCCCCCCCC    PPPPPPPPPPPP
NNN        NNN    CCCCCCCCCCCC    PPPPPPPPPPPP
NNN        NNN    CCC                PPP          PPP
NNN        NNN    CCC                PPP          PPP
NNN        NNN    CCC                PPP          PPP
NNNNNN     NNN    CCC                PPP          PPP
NNNNNN     NNN    CCC                PPP          PPP
NNNNNN     NNN    CCC                PPP          PPP
NNN    NNN NNN    CCC                PPPPPPPPPPPP
NNN    NNN NNN    CCC                PPPPPPPPPPPP
NNN    NNN NNN    CCC                PPPPPPPPPPPP
NNN    NNNNNN     CCC                PPP
NNN    NNNNNN     CCC                PPP
NNN    NNNNNN     CCC                PPP
NNN        NNN    CCC                PPP
NNN        NNN    CCC                PPP
NNN        NNN    CCC                PPP
NNN        NNN    CCCCCCCCCCCC       PPP
NNN        NNN    CCCCCCCCCCCC       PPP
NNN        NNN    CCCCCCCCCCCC       PPP
```

```
NN      NN  CCCCCCC PPPPPPP    SSSSSSSS TTTTTTTTTT  AAAAAA   MM      MM  OOOOOO  DDDDDDD
NN      NN  CCCCCCC PPPPPPP    SSSSSSSS TTTTTTTTTT  AAAAAA   MM      MM  OOOOOO  DDDDDDD
NN      NN  CC      PP    PP  SS            TT     AA    AA  MMMM  MMMM  OO    OO DD    DD
NNNN    NN  CC      PP    PP  SS            TT     AA    AA  MM MM MM MM OO    OO DD    DD
NNNN    NN  CC      PP    PP  SS            TT     AA    AA  MM  MM  MM  OO    OO DD    DD
NN NN   NN  CC      PPPPPPP    SSSSSS       TT     AA    AA  MM      MM  OO    OO DD    DD
NN NN   NN  CC      PPPPPPP    SSSSSS       TT     AAAAAAAA  MM      MM  OO    OO DD    DD
NN  NNNNNN  CC      PP             SS       TT     AAAAAAAA  MM      MM  OO    OO DD    DD
NN  NNNNNN  CC      PP             SS       TT     AA    AA  MM      MM  OO    OO DD    DD  ....
NN      NN  CC      PP             SS       TT     AA    AA  MM      MM  OO    OO DD    DD  ....
NN      NN  CCCCCCC PP        SSSSSSSS      TT     AA    AA  MM      MM  OOOOOO  DDDDDDD    ....
NN      NN  CCCCCCC PP        SSSSSSSS      TT     AA    AA  MM      MM  OOOOOO  DDDDDDD    ....
```

```
LL            IIIIII   SSSSSSSS
LL            !IIIII   SSSSSSSS
LL            II      SS
LL            II      SS
LL            II      SS
LL            II       SSSSSS
LL            II       SSSSSS
LL            II           SS
LL            II           SS
LL            II           SS
LL            II           SS
LLLLLLLLLL    IIIIII   SSSSSSSS
LLLLLLLLLL    IIIIII   SSSSSSSS
```

```
   1   0001  0  %TITLE  'Modules Configurator, Console, Loader, Looper Parsing'
   2   0002  0  MODULE NCPSTAMOD (IDENT = 'V04-000',LIST(NOOBJECT)) =
   3   0003  1  BEGIN
   4   0004  1
   5   0005  1
   6   0006  1  !*******************************************************************
   7   0007  1  !*                                                                 *
   8   0008  1  !*   COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                       *
   9   0009  1  !*   DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.        *
  10   0010  1  !*   ALL RIGHTS RESERVED.                                          *
  11   0011  1  !*                                                                 *
  12   0012  1  !*   THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  *
  13   0013  1  !*   ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE  *
  14   0014  1  !*   INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER  *
  15   0015  1  !*   COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  *
  16   0016  1  !*   OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY  *
  17   0017  1  !*   TRANSFERRED.                                                   *
  18   0018  1  !*                                                                 *
  19   0019  1  !*   THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE  *
  20   0020  1  !*   AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT  *
  21   0021  1  !*   CORPORATION.                                                   *
  22   0022  1  !*                                                                 *
  23   0023  1  !*   DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS  *
  24   0024  1  !*   SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.        *
  25   0025  1  !*                                                                 *
  26   0026  1  !*                                                                 *
  27   0027  1  !*******************************************************************
  28   0028  1  !
  29   0029  1
  30   0030  1  !++
  31   0031  1  ! FACILITY:     Network Control Program (NCP)
  32   0032  1
  33   0033  1  ! ABSTRACT:
  34   0034  1  !
  35   0035  1  !       States and data for the parsing of NCP Configurator module parameters
  36   0036  1
  37   0037  1  ! ENVIRONMENT:  VAX/VMS Operating System
  38   0038  1
  39   0039  1  ! AUTHOR:       Bob Grosso October 1982
  40   0040  1
  41   0041  1  ! MODIFIED BY:
  42   0042  1
  43   0043  1
  44   0044  1  !--
```

K 8

```
  46      0045  1 %SBTTL  'Definitions'
  47      0046  1
  48      0047  1 !
  49      0048  1 ! INCLUDE FILES:
  50      0049  1 !
  51      0050  1
  52      0051  1       LIBRARY 'LIB$:NMALIBRY';
  53      0052  1       LIBRARY 'LIB$:NCPLIBRY';
  54      0053  1       LIBRARY 'SYS$LIBRARY:TPAMAC';
  55      0054  1
  56      0055  1 !
  57      0056  1 ! EXTERNAL REFERENCES:
  58      0057  1 !
  59      0058  1
  60      0059  1       ACT_DFN                  ! Action routine externals
  61      0060  1
  62      0061  1 EXTERNAL
  63      0062  1       NCP$GL_QUALPRS;          ! Flag presence of qualifier
  64      0063  1
  65      0064  1 LITERAL
  66      0065  1       QUALPRESENT = 1;         ! Flag presence of qualifier on command line
  67      0066  1
```

```
    69        0067  1  %SBTTL  'Set Parameter blocks'
    70        0068  1  !
    71        0069  1  !
    72        0070  1  !          Set Configurator Parameter Blocks
    73        0071  1  !
    74        0072  1
    75      P 0073  1          BUILD_PCL
    76      P 0074  1
    77      P 0075  1          (MCF,                                          ! Module Configurator
    78      P 0076  1
    79      P 0077  1          CIR, TKN,         PCCN_CIR, ,
    80      P 0078  1          SUR, NUMB,        PCCN_SUR, ,
    81      P 0079  1
    82      P 0080  1          , END, , ,
    83      P 0081  1
    84      P 0082  1          )
    85        0083  1
    86        0084  1
    87        0085  1
    88      P 0086  1          BUILD_PBK
    89      P 0087  1
    90      P 0088  1          (MCF,                                          ! Module Configurator
    91      P 0089  1
    92      P 0090  1          CIR, TKN, , ,                                  ! Circuit is a qualifier
    93      P 0091  1          KCI, LITB, NMA$C_ENT_KNO, MCF_CIR,             ! Known circuits
    94      P 0092  1
    95      P 0093  1          SUR_ENAB, LITB, NMA$C_SUR_ENA, MCF_SUR,
    96      P 0094  1          SUR_DISAB, LITB, NMA$C_SUR_DIS, MCF_SUR,
    97      P 0095  1
    98        0096  1          )
    99        0097  1
   100        0098  1  BIND    PDB$G_MCF_ENT = UPLIT BYTE(0, %ASCIC 'CONFIGURATOR');
   101        0099  1
   102      P 0100  1          BUILD_SDB
   103        0101  1          (MCF, NMA$C_ENT_MOD, MCF_ENT, MCF)
```

```
: 105     0102  1 !
: 106     0103  1 !        Set Console Parameter Blocks
: 107     0104  1 !
: 108     0105  1
: 109   P 0106  1          BUILD_PCL
: 110   P 0107  1
: 111   P 0108  1          (MCS,                                ! Module Console
: 112   P 0109  1
: 113   P 0110  1          RTR, NUMB,      PCCO_RTR, ,
: 114   P 0111  1
: 115   P 0112  1          , END, , ,
: 116   P 0113  1          )
: 117     0114  1
: 118     0115  1
: 119     0116  1
: 120   P 0117  1          BUILD_PBK
: 121   P 0118  1
: 122   P 0119  1          (MCS,                                ! Module Console
: 123   P 0120  1
: 124   P 0121  1          RTR, NUMB, , ,
: 125   P 0122  1
: 126     0123  1          )
: 127     0124  1
: 128     0125  1 BIND     PDB$G_MCS_ENT = UPLIT BYTE(0, %ASCIC 'CONSOLE');
: 129     0126  1
: 130   P 0127  1          BUILD_SDB
: 131     0128  1          (MCS, NMA$C_ENT_MOD, MCS_ENT, MCS)
```

N 8

NCPSTAMOD   Modules Configurator, Console, Loader, Looper P 16-Sep-1984 00:46:46   VAX-11 Bliss-32 V4.0-742   Page 5
V04-000   Set Parameter blocks   14-Sep-1984 12:48:28   [NCP.SRC]NCPSTAMOD.B32;1   (5)

```
  133       0129   1 !
  134       0130   1 !      Set Loader Parameter Blocks
  135       0131   1 !
  136       0132   1
  137     P 0133   1        BUILD_PCL
  138     P 0134   1
  139     P 0135   1        (MLD,                                 ! Module Loader
  140     P 0136   1
  141     P 0137   1        ASS, NUMB,        PCLD_ASS, ,
  142     P 0138   1
  143     P 0139   1        , END, , ,
  144     P 0140   1
  145       0141   1        )
  146       0142   1
  147       0143   1
  148     P 0144   1        BUILD_PBK
  149     P 0145   1
  150     P 0146   1        (MLD,                                 ! Module Loader
  151     P 0147   1
  152     P 0148   1        ASS_ENAB, LITB, NMA$C_ASS_ENA, MLD_ASS,
  153     P 0149   1        ASS_DISAB, LITB, NMA$C_ASS_ENA, MLD_ASS,
  154     P 0150   1
  155       0151   1        )
  156       0152   1
  157       0153   1 BIND   PDB$G_MLD_ENT = UPLIT BYTE(0, %ASCIC 'LOADER');
  158       0154   1
  159     P 0155   1        BUILD_SDB
  160       0156   1        (MLD, NMA$C_ENT_MOD, MLD_ENT, MLD)
```

NCPSTAMOD   Modules Configurator, Console, Loader, Looper P 16-Sep-1984 00:46:46   VAX-11 Bliss-32 V4.0-742   Page 6
V04-000   Set Parameter blocks   14-Sep-1984 12:48:28   [NCP.SRC]NCPSTAMOD.B32;1   (6)

B 9

```
 162    0157  1 !
 163    0158  1 !              Set Looper Parameter Blocks
 164    0159  1 !
 165    0160  1
 166  P 0161  1              BUILD_PCL
 167  P 0162  1
 168  P 0163  1              (MLP,                                    ! Module Looper
 169  P 0164  1
 170  P 0165  1              ASS, NUMB,       PCLP_ASS, ,
 171  P 0166  1
 172  P 0167  1              , END, , ,
 173  P 0168  1
 174    0169  1              )
 175    0170  1
 176    0171  1
 177  P 0172  1              BUILD_PBK
 178  P 0173  1
 179  P 0174  1              (MLP,                                    ! Module Looper
 180  P 0175  1
 181  P 0176  1              ASS_ENAB, LITB, NMA$C_ASS_ENA, MLP_ASS,
 182  P 0177  1              ASS_DISAB, LITB, NMA$C_ASS_ENA, MLP_ASS,
 183  P 0178  1
 184    0179  1              )
 185    0180  1
 186    0181  1 BIND     PDB$G_MLP_ENT = UPLIT BYTE(0, %ASCIC 'LOOPER');
 187    0182  1
 188  P 0183  1              BUILD_SDB
 189    0184  1              (MLP, NMA$C_ENT_MOD, MLP_ENT, MLP)
```

C 9

NCPSTAMOD   Modules Configurator, Console, Loader, Looper P 16-Sep-1984 00:46:46   VAX-11 Bliss-32 V4.0-742   Page 7
V04-000   Clear Parameter blocks   14-Sep-1984 12:48:28   [NCP.SRC]NCPSTAMOD.B32;1   (7)

```
  191         0185  1 %SBTTL  'Clear Parameter blocks'
  192         0186  1
  193         0187  1 !
  194         0188  1 !        Clear Configurator Parameter Blocks
  195         0189  1 !
  196         0190  1
  197         0191  1
  198       P 0192  1          BUILD_PCL
  199       P 0193  1
  200       P 0194  1          (CCF,                                      ! Module Configurator
  201       P 0195  1
  202       P 0196  1          CIR, TKN,       PCCN_CIR, ,
  203       P 0197  1
  204       P 0198  1          SUR, LITB,      PCCN_SUR, ,
  205       P 0199  1
  206       P 0200  1          , END, , ,
  207       P 0201  1
  208         0202  1          )
  209         0203  1
  210       P 0204  1          BUILD_PBK
  211       P 0205  1
  212       P 0206  1          (CCF,                                      ! Module Configurator
  213       P 0207  1
  214       P 0208  1          ALL, LITB, 0, VRB_ALL,
  215       P 0209  1          CIR, TKN, 0, ,                             ! Circuit is a qualifier
  216       P 0210  1          KCI, LITB, NMA$C_ENT_KNO, CCF_CIR,         ! Known circuits
  217       P 0211  1
  218       P 0212  1          SUR, LITB, 0, ,
  219       P 0213  1
  220         0214  1          )
  221         0215  1
  222       P 0216  1          BUILD_SDB
  223       P 0217  1
  224         0218  1          (CCF, NMA$C_ENT_MOD, MCF_ENT, CCF)
  225         0219  1
  226         0220  1
```

```
 228    0221  1 !
 229    0222  1 !           Clear Console Parameter Blocks
 230    0223  1 !
 231    0224  1
 232    0225  1
 233  P 0226  1           BUILD_PCL
 234  P 0227  1
 235  P 0228  1           (CCS,                                      ! Module Console
 236  P 0229  1
 237  P 0230  1           RTR, LITB,      PCCO_RTR, ,
 238  P 0231  1
 239  P 0232  1           , END, , ,
 240  P 0233  1
 241    0234  1           )
 242    0235  1
 243  P 0236  1           BUILD_PBK
 244  P 0237  1
 245  P 0238  1           (CCS,                                      ! Module Console
 246  P 0239  1
 247  P 0240  1           ALL, LITB, 0, VRB_ALL,
 248  P 0241  1
 249  P 0242  1           RTR, LITB, 0, ,
 250  P 0243  1
 251    0244  1           )
 252    0245  1
 253  P 0246  1           BUILD_SDB
 254  P 0247  1
 255    0248  1           (CCS, NMA$C_ENT_MOD, MCS_ENT, CCS)
 256    0249  1
 257    0250  1
```

```
259    0251  1  !
260    0252  1  !        Clear Loader Parameter Blocks
261    0253  1  !
262    0254  1
263    0255  1
264  P 0256  1        BUILD_PCL
265  P 0257  1
266  P 0258  1        (CLD,                                    ! Module Loader
267  P 0259  1
268  P 0260  1
269  P 0261  1        ASS, LITB,      PCLD_ASS, ,
270  P 0262  1
271  P 0263  1        , END, , ,
272  P 0264  1        )
273    0265  1
274    0266  1
275  P 0267  1        BUILD_PBK
276  P 0268  1
277  P 0269  1        (CLD,                                    ! Module Loader
278  P 0270  1
279  P 0271  1        ALL, LITB, 0, VRB_ALL,
280  P 0272  1
281  P 0273  1        ASS, LITB, 0, ,
282  P 0274  1        )
283    0275  1
284    0276  1
285  P 0277  1        BUILD_SDB
286  P 0278  1
287    0279  1        (CLD, NMA$C_ENT_MOD, MLD_ENT, CLD)
288    0280  1
289    0281  1
```

```
291    0282  1 !
292    0283  1 !      Clear Looper Parameter Blocks
293    0284  1 !
294    0285  1
295    0286  1
296  P 0287  1        BUILD_PCL
297  P 0288  1
298  P 0289  1        (CLP,                                  ! Module Looper
299  P 0290  1
300  P 0291  1        ASS, LITB.       PCLP_ASS, ,
301  P 0292  1
302  P 0293  1        , END, , ,
303  P 0294  1
304    0295  1        )
305    0296  1
306  P 0297  1        BUILD_PBK
307  P 0298  1
308  P 0299  1        (CLP,                                  ! Module Looper
309  P 0300  1
310  P 0301  1        ALL, LITB, 0, VRB_ALL,
311  P 0302  1
312  P 0303  1        ASS, LITB, 0, ,
313  P 0304  1
314    0305  1        )
315    0306  1
316  P 0307  1        BUILD_SDB
317  P 0308  1
318    0309  1        (CLP, NMASC_ENT_MOD, MLP_ENT, CLP)
319    0310  1
320    0311  1
```

```
322        0312  1 %SBTTL  'Prompt strings'
323        0313  1 !
324        0314  1 !
325        0315  1 !      Build prompt strings
326        0316  1 !
327        0317  1 !
328        0318  1 BIND
329        0319  1
330      P 0320  1        PROMPT_STRINGS
331      P 0321  1        (MCF,
332      P 0322  1
333      P 0323  1        DAT, '                 (CIRCUIT name, or KNOWN): ',
334      P 0324  1        KWN, '                               (CIRCUITS): ',
335      P 0325  1        SUR, 'Surveillance flag  (ENABLED, DISABLED): ',
336      P 0326  1
337        0327  1        ),
338        0328  1
339      P 0329  1        PROMPT_STRINGS
340      P 0330  1        (CCF,
341      P 0331  1
342      P 0332  1        DAT, '                 (CIRCUIT name, or KNOWN): ',
343      P 0333  1        KWN, '                               (CIRCUITS): ',
344      P 0334  1        ALL, 'All Configurator parameters     (Y, N): ',
345        0335  1
346      P 0336  1        SUR, 'Surveillance flag               (Y, N): ',
347      P 0337  1
348        0338  1        );
349        0339  1
350        0340  1 !
351        0341  1 !      Module Console prompts
352        0342  1 !
353        0343  1 BIND
354        0344  1
355      P 0345  1        PROMPT_STRINGS
356      P 0346  1        (MCS,
357      P 0347  1
358      P 0348  1        RTR, 'Reservation timer    (1-65535 seconds): ',
359      P 0349  1
360        0350  1        ),
361        0351  1
362      P 0352  1        PROMPT_STRINGS
363      P 0353  1        (CCS,
364      P 0354  1
365      P 0355  1        ALL, 'All Console parameters          (Y, N): ',
366      P 0356  1
367      P 0357  1        RTR, 'Reservation timer               (Y, N): ',
368      P 0358  1
369        0359  1        );
370        0360  1
371        0361  1 !
372        0362  1 !      Module Loader prompts
373        0363  1 !
374        0364  1 BIND
375        0365  1
376      P 0366  1        PROMPT_STRINGS
377      P 0367  1        (MLD,
378      P 0368  1
```

```
  379      P 0369  1            ASS, 'Assistance control (ENABLED, DISABLED): ',
  380      P 0370  1
  381        0371  1            ),
  382        0372  1
  383      P 0373  1            PROMPT_STRINGS
  384      P 0374  1            (CLD,
  385      P 0375  1
  386      P 0376  1            ALL, 'All Loader parameters         (Y, N): ',
  387      P 0377  1
  388      P 0378  1            ASS, 'Assistance control            (Y, N): ',
  389      P 0379  1
  390        0380  1            );
  391        0381  1
  392        0382  1    !
  393        0383  1    !       Module Looper prompts
  394        0384  1    !
  395        0385  1 BIND
  396        0386  1
  397      P 0387  1            PROMPT_STRINGS
  398      P 0388  1            (MLP,
  399      P 0389  1
  400      P 0390  1            ASS, 'Assistance control (ENABLED, DISABLED): ',
  401      P 0391  1
  402        0392  1            ),
  403        0393  1
  404      P 0394  1            PROMPT_STRINGS
  405      P 0395  1            (CLP,
  406      P 0396  1
  407      P 0397  1            ALL, 'All Looper parameters         (Y, N): ',
  408      P 0398  1
  409      P 0399  1            ASS, 'Assistance control            (Y, N): ',
  410      P 0400  1
  411        0401  1            );
```

I 9

NCPSTAMOD   Modules Configurator, Console, Loader, Looper P 16-Sep-1984 00:46:46   VAX-11 Bliss-32 V4.0-742   Page 13
V04-000   Declare entry points to TPARSE tables   14-Sep-1984 12:48:28   [NCP.SRC]NCPSTAMOD.B32;1   (12)

```
  413        0402  1  %SBTTL  'Declare entry points to TPARSE tables'
  414        0403  1
  415        0404  1  !
  416        0405  1  !        Declare entry points to this TPARSE table
  417        0406  1  !
  418        0407  1
  419        0408  1  $INIT_STATE (NCP$G_STTBL_MOD, NCP$G_KYTBL_MOD);
  420        0409  1
  421        0410  1  FORWARD
  422        0411  1              ST_MCF:              VECTOR [0],     ! Set Module Configurator
  423        0412  1              ST_CCF:              VECTOR [0],     ! Clear Module Configurator
  424        0413  1              ST_MCS:              VECTOR [0],     ! Set Module Console
  425        0414  1              ST_CCS:              VECTOR [0],     ! Clear Module Console
  426        0415  1              ST_MLD:              VECTOR [0],     ! Set Module Loader
  427        0416  1              ST_CLD:              VECTOR [0],     ! Clear Module Loader
  428        0417  1              ST_MLP:              VECTOR [0],     ! Set Module Looper
  429        0418  1              ST_CLP:              VECTOR [0];     ! Clear Module Looper
  430        0419  1
  431        0420  1
  432        0421  1  GLOBAL BIND
  433        0422  1              NCP$G_STTBL_MODCNF = ST_MCF,
  434        0423  1              NCP$G_KYTBL_MODCNF = NCP$G_KYTBL_MOD,
  435        0424  1              NCP$G_STTBL_CCF = ST_CCF,
  436        0425  1              NCP$G_KYTBL_CCF = NCP$G_KYTBL_MOD,
  437        0426  1
  438        0427  1              NCP$G_STTBL_MODCNS = ST_MCS,
  439        0428  1              NCP$G_KYTBL_MODCNS = NCP$G_KYTBL_MOD,
  440        0429  1              NCP$G_STTBL_CCS = ST_CCS,
  441        0430  1              NCP$G_KYTBL_CCS = NCP$G_KYTBL_MOD,
  442        0431  1
  443        0432  1              NCP$G_STTBL_MODLOA = ST_MLD,
  444        0433  1              NCP$G_KYTBL_MODLOA = NCP$G_KYTBL_MOD,
  445        0434  1              NCP$G_STTBL_CLD = ST_CLD,
  446        0435  1              NCP$G_KYTBL_CLD = NCP$G_KYTBL_MOD,
  447        0436  1
  448        0437  1              NCP$G_STTBL_MODLOO = ST_MLP,
  449        0438  1              NCP$G_KYTBL_MODLOO = NCP$G_KYTBL_MOD,
  450        0439  1              NCP$G_STTBL_CLP = ST_CLP,
  451        0440  1              NCP$G_KYTBL_CLP = NCP$G_KYTBL_MOD;
```

```
453        0441   1    %SBTTL   'SET Configurator Module Parameters'
454        0442   1
455        0443   1
456        0444   1    |
457        0445   1    |        SET/DEFINE MODULE Configurator parameter states
458        0446   1
459        0447   1
460    P   0448   1    $STATE   (ST_MCF,
461    P   0449   1             (TPA$_EOS),
462    P   0450   1             (TPA$_LAMBDA, ST_MCF_DAT)
463        0451   1             );
464        0452   1
465    P   0453   1    $STATE   (
466        0454   1             (TPA$_LAMBDA, , ACT$PRMPT, , , PMT$G_MCF_DAT));
467        0455   1
468        0456   1    |
469        0457   1    |        Configurator is qualified by Circuit or Known Circuits
470        0458   1    |
471    P   0459   1    $STATE   (ST_MCF_DAT,
472    P   0460   1             ('CIRCUIT')
473    P   0461   1             ('KNOWN', ST_MCF_DAT_KWN),
474    P   0462   1             (TPA$_EOS, ST_MCF_PMT_CIR, ACT$SAVPRM, , ,PBK$G_MCF_KCI),
475        0463   1             );
476        0464   1
477    P   0465   1    $STATE   (
478    P   0466   1             ((SE_CIRC_ID), ST_MCF_PMT_CIR, ACT$SAVPRM,
479        0467   1                     QUALPRESENT, NCP$GL_QUALPRS, PBK$G_MCF_CIR));
480        0468   1
481    P   0469   1    $STATE   (ST_MCF_DAT_KWN,
482        0470   1             (TPA$_LAMBDA));
483        0471   1
484    P   0472   1             COMMAND_PROMPT
485    P   0473   1             (MCF, KWN, NCP$_INVKEY,
486        0474   1
487    P   0475   1             ('CIRCUITS', ST_MCF_PMT_CIR, ACT$SAVPRM, , ,PBK$G_MCF_KCI),
488        0476   1             )
489        0477   1    |
490        0478   1    |        Prompt for circuit parameters
491        0479   1    |
492    P   0480   1    $STATE   (ST_MCF_PMT_CIR,
493    P   0481   1             (TPA$_EOS),                     | start prompting if EOS
494        0482   1             (TPA$_LAMBDA, ST_MCF_PRC));     | Else try parsing parameters
495        0483   1
496    P   0484   1             PROMPT_STATES
497    P   0485   1             (MCF,
498    P   0486   1
499        0487   1             SUR)
500        0488   1
501        0489   1
502    P   0490   1    $STATE   (ST_MCF_DOIT,
503    P   0491   1             (TPA$_EOS, TPA$_EXIT, ACT$VRB_UTILITY, , , SDB$G_MCF),
504        0492   1             );
505        0493   1
```

```
   507    P 0494  1  $STATE   (ST_MCF_PRC,
   508    P 0495  1            ((SE_ALC), ST_MCF_DOIT),
   509    P 0496  1
   510    P 0497  1            DISPATCH_STATES
   511    P 0498  1            (MCF,
   512    P 0499  1
   513    P 0500  1            SUR, 'SURVEILLANCE',
   514    P 0501  1
   515    P 0502  1            )
   516    P 0503  1
   517    P 0504  1            ,(TPA$_EOS, ST_MCF_DOIT)
   518      0505  1            );
   519      0506  1
   520    P 0507  1  $STATE   (ST_MCF_SUR,
   521    P 0508  1
   522    P 0509  1            KEYWORD_STATE
   523    P 0510  1            (MCF,
   524    P 0511  1
   525    P 0512  1            SUR_ENAB,        'ENABLED'
   526    P 0513  1            SUR_DISAB,       'DISABLED',
   527    P 0514  1
   528      0515  1            ));
   529      0516  1
   530      0517  1  !
   531      0518  1  !        Process States
   532      0519  1  !
   533      0520  1            PROCESS_STATES
   534    P 0521  1            (MCF,
   535    P 0522  1
   536    P 0523  1            SUR ,
   537    P 0524  1
   538      0525  1            )
   539      0526  1
```

```
541        0527  1    %SBTTL  'SET Console Module Parameters'
542        0528  1
543        0529  1    !
544        0530  1    !       SET/DEFINE MODULE Console parameter states
545        0531  1    !
546        0532  1
547        0533  1
548      P 0534  1    $STATE  (ST_MCS,
549      P 0535  1             (TPA$_EOS),
550      P 0536  1             (TPA$_LAMBDA, ST_MCS_PRC)
551        0537  1             );
552        0538  1
553        0539  1
554      P 0540  1             PROMPT_STATES
555      P 0541  1             (MCS,
556      P 0542  1
557        0543  1             RTR)
558        0544  1
559        0545  1
560      P 0546  1    $STATE  (ST_MCS_DOIT,
561      P 0547  1             (TPA$_EOS, TPA$_EXIT, ACT$VRB_UTILITY, , , SDB$G_MCS),
562        0548  1             );
563        0549  1
564        0550  1
565      P 0551  1    $STATE  (ST_MCS_PRC,
566      P 0552  1             ((SE_ALL), ST_MCS_DOIT),
567      P 0553  1
568      P 0554  1             DISPATCH_STATES
569      P 0555  1             (MCS,
570      P 0556  1
571      P 0557  1             RTR, 'RESERVATION',
572      P 0558  1
573      P 0559  1             )
574      P 0560  1
575      P 0561  1             ,(TPA$_EOS, ST_MCS_DOIT)
576        0562  1             );
577        0563  1
578        0564  1    !
579        0565  1    !       Process States
580        0566  1    !
581      P 0567  1             PROCESS_STATES
582      P 0568  1             (MCS,
583      P 0569  1
584      P 0570  1             RTR, 'TIMER',
585      P 0571  1
586        0572  1             )
587        0573  1
588        0574  1    !
589        0575  1    !       Subexpression states
590        0576  1    !
591      P 0577  1             SUB_EXPRESSIONS
592      P 0578  1             (MCS,
593      P 0579  1
594      P 0580  1             RTR, TPA$_DECIMAL,
595      P 0581  1
596        0582  1             )
597        0583  1
```

```
  599    0584  1
  600    0585  1   %SBTTL   'SET Loader Module Parameters'
  601    0586  1
  602    0587  1   |
  603    0588  1   |         SET/DEFINE Loader parameter states
  604    0589  1   |
  605    0590  1
  606  P 0591  1   $STATE   (ST_MLD,
  607  P 0592  1             (TPA$_EOS),
  608  P 0593  1             (TPA$_LAMBDA, ST_MLD_PRC)
  609    0594  1             );
  610    0595  1
  611    0596  1
  612  P 0597  1             PROMPT_STATES
  613  P 0598  1             (MLD,
  614  P 0599  1
  615    0600  1             ASS)
  616    0601  1
  617    0602  1
  618  P 0603  1   $STATE   (ST_MLD_DOIT,
  619  P 0604  1             (TPA$_EOS, TPA$_EXIT, ACT$VRB_UTILITY, , , SDB$G_MLD),
  620    0605  1             );
  621    0606  1
  622    0607  1
  623  P 0608  1   $STATE   (ST_MLD_PRC,
  624  P 0609  1             ((SE_ALL), ST_MLD_DOIT),
  625  P 0610  1
  626  P 0611  1             DISPATCH_STATES
  627  P 0612  1             (MLD,
  628  P 0613  1
  629  P 0614  1             ASS, 'ASSISTANCE',
  630  P 0615  1
  631  P 0616  1             )
  632  P 0617  1
  633  P 0618  1             ,(TPA$_EOS, ST_MLD_DOIT)
  634    0619  1             );
  635    0620  1
  636    0621  1   |
  637    0622  1   |         Process States
  638    0623  1   |
  639    0624  1             PROCESS_STATES
  640  P 0625  1             (MLD,
  641  P 0626  1
  642  P 0627  1             ASS, .
  643  P 0628  1
  644    0629  1             )
  645    0630  1
  646  P 0631  1   $STATE   (ST_MLD_ASS,
  647  P 0632  1
  648  P 0633  1             KEYWORD_STATE
  649  P 0634  1             (MLD,
  650    0635  1
  651  P 0636  1             ASS_ENAB,          'ENABLED'
  652  P 0637  1             ASS_DISAB,         'DISABLED',
  653  P 0638  1
  654    0639  1             ));
```

```
 656      0640  1
 657      0641  1      %SBTTL   'SET Looper Module Parameters'
 658      0642  1
 659      0643  1      !
 660      0644  1      !          SET/DEFINE Looper parameter states
 661      0645  1      !
 662      0646  1
 663    P 0647  1      $STATE   (ST_MLP,
 664    P 0648  1               (TPA$_EOS),
 665    P 0649  1               (TPA$_LAMBDA, ST_MLP_PRC)
 666      0650  1               );
 667      0651  1
 668      0652  1
 669    P 0653  1               PROMPT_STATES
 670    P 0654  1               (MLP,
 671    P 0655  1
 672      0656  1               ASS)
 673      0657  1
 674      0658  1
 675    P 0659  1      $STATE   (ST_MLP_DOIT,
 676    P 0660  1               (TPA$_EOS, TPA$_EXIT, ACT$VRB_UTILITY, . , SDB$G_MLP),
 677      0661  1               );
 678      0662  1
 679      0663  1
 680    P 0664  1      $STATE   (ST_MLP_PRC,
 681    P 0665  1               ((SE_ALL), ST_MLP_DOIT),
 682    P 0666  1
 683    P 0667  1               DISPATCH_STATES
 684    P 0668  1               (MLP,
 685    P 0669  1
 686    P 0670  1               ASS, 'ASSISTANCE',
 687    P 0671  1
 688    P 0672  1               )
 689    P 0673  1
 690    P 0674  1               ,(TPA$_EOS, ST_MLP_DOIT)
 691      0675  1               );
 692      0676  1
 693      0677  1      !
 694      0678  1      !          Process States
 695      0679  1      !
 696    P 0680  1               PROCESS_STATES
 697    P 0681  1               (MLP,
 698    P 0682  1
 699    P 0683  1               ASS, .
 700    P 0684  1
 701      0685  1               )
 702      0686  1
 703    P 0687  1      $STATE   (ST_MLP_ASS,
 704    P 0688  1
 705    P 0689  1               KEYWORD_STATE
 706    P 0690  1               (MLP,
 707    P 0691  1
 708    P 0692  1               ASS_ENAB,        'ENABLED',
 709    P 0693  1               ASS_DISAB,       'DISABLED',
 710    P 0694  1
 711      0695  1               ));
```

B 10

NCPSTAMOD   Modules Configurator, Console, Loader, Looper P 16-Sep-1984 00:46:46   VAX-11 Bliss-32 V4.0-742   Page 19
V04-000   CLEAR Configurator Module Parameters   14-Sep-1984 12:48:28   [NCP.SRC]NCPSTAMOD.B32;1   (18)

```
  713      0696  1  %SBTTL  'CLEAR Configurator Module Parameters'
  714      0697  1
  715      0698  1  !
  716      0699  1  !          CLEAR/PURGE MODULE Configurator parameter states
  717      0700  1  !
  718      0701  1
  719    P 0702  1  $STATE  (ST_CCF,
  720    P 0703  1          (TPA$_EOS),
  721    P 0704  1          (TPA$_LAMBDA, ST_CCF_DAT)
  722      0705  1          );
  723      0706  1
  724    P 0707  1  $STATE  (,
  725      0708  1          (TPA$_LAMBDA, , ACT$PRMPT, , , PMT$G_CCF_DAT));
  726      0709  1
  727      0710  1  !
  728      0711  1  !          Configurator is qualified by Circuit or Known Circuits
  729      0712  1  !
  730    P 0713  1  $STATE  (ST_CCF_DAT,
  731    P 0714  1          ('CIRCUIT'),
  732    P 0715  1          ('KNOWN', ST_CCF_DAT_KWN),
  733    P 0716  1          (TPA$_EOS, ST_CCF_PMT_CIR, ACT$SAVPRM, , ,PBK$G_CCF_KCI),
  734      0717  1          );
  735      0718  1
  736    P 0719  1  $STATE  (,
  737    P 0720  1          ((SE_CIRC_ID), ST_CCF_PMT_CIR, ACT$SAVPRM,
  738      0721  1                  QUALPRESENT, NCP$GL_QUALPRS, PBK$G_CCF_CIR));
  739      0722  1
  740    P 0723  1  $STATE  (ST_CCF_DAT_KWN,
  741      0724  1          (TPA$_LAMBDA));
  742      0725  1
  743    P 0726  1          COMMAND_PROMPT
  744    P 0727  1          (CCF, KWN, NCP$_INVKEY,
  745    P 0728  1
  746    P 0729  1          ('CIRCUITS', ST_CCF_PMT_CIR, ACT$SAVPRM, , ,PBK$G_CCF_KCI),
  747      0730  1          )
  748      0731  1  !
  749      0732  1  !          Prompt for circuit parameters
  750      0733  1  !
  751    P 0734  1  $STATE  (ST_CCF_PMT_CIR,
  752    P 0735  1          (TPA$_EOS),                        ! start prompting if EOS
  753      0736  1          (TPA$_LAMBDA, ST_CCF_PRC));        ! Else try parsing parameters
  754      0737  1
  755    P 0738  1          QUERY_STATES
  756    P 0739  1          (CCF,
  757    P 0740  1
  758      0741  1          ALL, SUR)
  759      0742  1
  760      0743  1
  761    P 0744  1  $STATE  (ST_CCF_DOIT,
  762    P 0745  1          (TPA$_EOS, TPA$_EXIT, ACT$VRB_UTILITY, , , SDB$G_CCF),
  763      0746  1          );
  764      0747  1
```

```
766        P 0748  1  $STATE  (ST_CCF_PRC,
767        P 0749  1             ((SE_ALL), ST_CCF_DOIT),
768        P 0750  1
769        P 0751  1             DISPATCH_STATES
770        P 0752  1             (CCF,
771        P 0753  1
772        P 0754  1             SUR, 'SURVEILLANCE',
773        P 0755  1
774        P 0756  1             )
775        P 0757  1
776        ? 0758  1             ,(TPAS_EOS, ST_CCF_DOIT)
777          0759              );
778          0760  1
779          0761  1
780          0762  1  !
781          0763  1  !        Process States
782          0764  1  !
783        P 0765  1           PROCESS_STATES
784        P 0766  1           (CCF,
785        P 0767  1
786        P 0768  1           SUR ,
787        P 0769  1
788          0770  1           )
789          0771  1
790          0772  1  !
791          0773  1  !        Subexpression states
792          0774  1  !
793          0775  1
794        P 0776  1           SUB_EXPRESSIONS
795        P 0777  1           (CCF,
796        P 0778  1
797        P 0779  1           ALL, TPAS_EOS,
798        P 0780  1           SUR, TPAS_LAMBDA,
799        P 0781  1
800          0782  1           )
```

```
 802    0783  1 %SBTTL  'CLEAR Console Module Parameters'
 803    0784  1
 804    0785  1 !
 805    0786  1 !        CLEAR/PURGE MODULE Console parameter states
 806    0787  1 !
 807    0788  1
 808  P 0789  1 $STATE  (ST_CCS,
 809  P 0790  1          (TPA$_EOS),
 810  P 0791  1          (TPA$_LAMBDA, ST_CCS_PRC)
 811    0792  1          );
 812    0793  1
 813    0794  1
 814  P 0795  1          QUERY_STATES
 815  P 0796  1          (CCS,
 816  P 0797  1
 817    0798  1          ALL, RTR)
 818    0799  1
 819    0800  1
 820  P 0801  1 $STATE  (ST_CCS_DOIT,
 821  P 0802  1          (TPA$_EOS, TPA$_EXIT, ACT$VRB_UTILITY, , , SDB$G_CCS),
 822    0803  1          );
 823    0804  1
 824    0805  1
 825  P 0806  1 $STATE  (ST_CCS_PRC,
 826  P 0807  1          ((SE_ALL), ST_CCS_DOIT),
 827  P 0808  1
 828  P 0809  1          DISPATCH_STATES
 829  P 0810  1          (CCS,
 830  P 0811  1
 831  P 0812  1          RTR, 'RESERVATION',
 832  P 0813  1
 833  P 0814  1          )
 834  P 0815  1
 835  P 0816  1          ,(TPA$_EOS, ST_CCS_DOIT)
 836    0817  1          );
 837    0818  1
 838    0819  1
 839    0820  1
 840    0821  1 !        Process States
 841    0822  1 !
 842  P 0823  1          PROCESS_STATES
 843  P 0824  1          (CCS,
 844  P 0825  1
 845  P 0826  1          RTR, 'TIMER',
 846  P 0827  1
 847    0828  1          )
 848    0829  1
 849    0830  1 !
 850    0831  1 !        Subexpression states
 851    0832  1 !
 852    0833  1
 853  P 0834  1          SUB_EXPRESSIONS
 854  P 0835  1          (CCS,
 855    0836  1
 856  P 0837  1          ALL, TPA$_EOS,
 857  P 0838  1          RTR, TPA$_LAMBDA,
 858  P 0839  1
```

; 859   0840 1   )

F 10

NCPSTAMOD   Modules Configurator, Console, Loader, Looper P 16-Sep-1984 00:46:46   VAX-11 Bliss-32 V4.0-742   Page 23
V04-000   CLEAR Loader Module Parameters   14-Sep-1984 12:48:28   [NCP.SRC]NCPSTAMOD.B32;1   (21)

```
861        0841  1  %SBTTL  'CLEAR Loader Module Parameters'
862        0842  1
863        0843  1  !
864        0844  1  !          CLEAR/PURGE MODULE Loader parameter states
865        0845  1  !
866        0846  1
867   P    0847  1  $STATE  (ST_CLD,
868   P    0848  1           (TPA$_EOS),
869   P    0849  1           (TPA$_LAMBDA, ST_CLD_PRC)
870        0850  1           ):
871        0851  1
872        0852  1
873   P    0853  1           QUERY_STATES
874   P    0854  1           (CLD,
875   P    0855  1
876        0856  1           ALL, ASS)
877        0857  1
878        0858  1
879   P    0859  1  $STATE  (ST_CLD_DOIT,
880   P    0860  1           (TPA$_EOS, TPA$_EXIT, ACT$VRB_UTILITY, , , SDB$G_CLD),
881        0861  1           ):
882        0862  1
883        0863  1
884   P    0864  1  $STATE  (ST_CLD_PRC,
885   P    0865  1           ((SE_ALL), ST_CLD_DOIT),
886        0866  1
887   P    0867  1           DISPATCH_STATES
888   P    0868  1           (CLD,
889   P    0869  1
890   P    0870  1           ASS, 'ASSISTANCE',
891   P    0871  1
892   P    0872  1           )
893   P    0873  1
894   P    0874  1           ,(TPA$_EOS, ST_CLD_DOIT)
895        0875  1           ):
896        0876  1
897        0877  1
898        0878  1  !
899        0879  1          Process States
900        0880  1  !
901   P    0881  1          PROCESS_STATES
902   P    0882  1          (CLD,
903   P    0883  1
904   P    0884  1          ASS, .
905   P    0885  1
906        0886  1          )
907        0887  1
908        0888  1  !
909        0889  1          Subexpression states
910        0890  1  !
911        0891  1
912   P    0892  1          SUB_EXPRESSIONS
913   P    0893  1          (CLD,
914   P    0894  1
915   P    0895  1          ALL, TPA$_EOS,
916   P    0896  1          ASS, TPA$_LAMBDA,
917   P    0897  1
```

: 918   0898 1   )

```
 920       0899  1  %SBTTL  'CLEAR Looper Module Parameters'
 921       0900  1  !
 922       0901  1  !
 923       0902  1  !            CLEAR/PURGE MODULE Looper parameter states
 924       0903  1  !
 925       0904  1
 926    P  0905  1  $STATE  (ST_CLP,
 927    P  0906  1           (TPA$_EOS),
 928    P  0907  1           (TPA$_LAMBDA, ST_CLP_PRC)
 929       0908  1           );
 930       0909  1
 931       0910  1
 932    P  0911  1           QUERY_STATES
 933    P  0912  1           (CLP,
 934    P  0913  1
 935       0914  1           ALL, ASS)
 936       0915  1
 937       0916  1
 938    P  0917  1  $STATE  (ST_CLP_DOIT,
 939    P  0918  1           (TPA$_EOS, TPA$_EXIT, ACT$VRB_UTILITY, , , SDB$G_CLP),
 940       0919  1           );
 941       0920  1
 942       0921  1
 943    P  0922  1  $STATE  (ST_CLP_PRC,
 944    P  0923  1           ((SE_ALL), ST_CLP_DOIT),
 945    P  0924  1
 946    P  0925  1           DISPATCH_STATES
 947    P  0926  1           (CLP,
 948    P  0927  1
 949    P  0928  1           ASS, 'ASSISTANCE',
 950    P  0929  1
 951    P  0930  1           )
 952    P  0931  1
 953    P  0932  1           ,(TPA$_EOS, ST_CLP_DOIT)
 954       0933  1           );
 955       0934  1
 956       0935  1
 957       0936  1  !
 958       0937  1  !          Process States
 959       0938  1  !
 960    P  0939  1           PROCESS_STATES
 961    P  0940  1           (CLP,
 962    P  0941  1
 963    P  0942  1           ASS, .
 964    P  0943  1
 965       0944  1           )
 966       0945  1
 967       0946  1  !
 968       0947  1  !          Subexpression states
 969       0948  1  !
 970       0949  1
 971    P  0950  1           SUB_EXPRESSIONS
 972    P  0951  1           (CLP,
 973    P  0952  1
 974    P  0953  1           ALL, TPA$_EOS,
 975    P  0954  1           ASS, TPA$_LAMBDA,
 976    P  0955  1
```

: 977   0956 1   )

```
  979      0957  1 %SBTTL  'Define Subexpressions'
  980      0958  1
  981      0959  1 !
  982      0960  1 !       Define Subexpressions from Library
  983      0961  1 !
  984      0962  1
  985      0963  1         SEM_ALL                          ! All parameter
  986      0964  1         SEM_CIRC_ID                      ! Circuit name
  987      0965  1         SEM_LINE_ID
  988      0966  1         SEM_QUERY                        ! Query state subexpressions
```

```
: 990     0967  1 END
: 991     0968  0 ELUDOM
```

NCPSTAMAC
LIS

NCPSTAMSE
LIS

NCPSTAMTR
LIS

NCPSTAMOD
LIS

NCPSTALOA
LIS

NCPSTALOO
LIS

NCPSTAMPR
LIS

NCPSTALOG
LIS